

Zadanie 02

Główne składowe odzieży

Celem zadania będzie przetestowanie poznanej techniki PCA w kontakcie z prawdziwymi danymi.

Dane do wykorzystania

Do realizacji zadania potrzebny będzie nam niewielki zbiór z fotografiami. Co prawda fotografie nie są danymi czysto wektorowymi (i ich praktyczne przetwarzanie wymaga brania pod uwagę również informacji o wzajemnym położeniu poszczególnych pikseli) - ale za to łatwo je interpretować metodą "na oko" i pozwalają na uzyskiwanie atrakcyjnych wizualnie rezultatów. Przy tworzeniu zbioru można współpracować z innymi studentami z kursu, ale w grupach po kilka osób (e.g. nie jest tak, że cały rocznik korzysta z pracy jednej osoby ;)).

Potrzebujemy zgromadzić zbiór zawierający 3 rodzaje różnych (ale nie diametralnie różnych) obiektów i zawierający przynajmniej po kilkanaście (lub więcej) przykładów każdego z nich (np. 3 typy zawartości i 20 zdjęć każdego z nich, łącznie 60 zdjęć). Postarajmy się też, by obiekty były wycentrowane oraz zaprezentowane na jednolitym tle i w podobnym oświetleniu. Co się nadaje?

- Trzy typy ubrań (np. koszule, t-shirty i marynarki) z katalogu ulubionej marki odzieżowej (prezentowane w tej samej pozie - albo po prostu bez modela).
 - Osoby potrzebujące oszczędzić na czasie mogą wykorzystać wybrane 3 klasy ze zbioru FashionMNIST (<https://github.com/zalandoresearch/fashion-mnist>). Wyniki będą mniej ciekawe (i realistyczne), bo FashionMNIST zawiera fotografie o niskiej rozdzielczości, za to perfekcyjnie przebrane i wycentrowane. Czyli dokładnie odwrotnie niż zwykle się zdarza. ;]
- Trzy rodzaje butów (np. sandały, trampki, trampki) z katalogów obuwniczych - tak jak w poprzednim punkcie, staramy się skorzystać z ujęć robionych pod tym samym kątem.
- Trzy rodzaje naczyń (np. kubki, szklanki, kieliszki) z katalogów odpowiedniego sklepu.
- Trzy rodzaje sztućców (np. łyżki, noże i widelce).
- Wygenerowane twarze trzech typów postaci (np. stereotypowy człek, elf i krzat) w grze komputerowej pozwalające na losowe generowanie wyglądu bohaterów.
 - Zadanie miało być pierwotnie oparte o właśnie taki zbiór, ale nie znalazłem darmowej gry o niskich wymagach sprzętowych i satysfakcjonującej jakości generatora.

Wstępna obróbka

Po zgromadzeniu odpowiedniej liczby zdjęć przygotowujemy je do późniejszego użycia. W tym celu można użyć np. biblioteki Pillow (<https://pillow.readthedocs.io> - do pracy z obrazami) i biblioteki numpy (<https://note.nkml.me/en/python-numpy-image-processing> - do pracy z macierzami i wektorami) - lub dowolnego innego preferowanego narzędzia.

- By przyspieszyć późniejsze obliczenia zmniejszamy rozdzielczość fotografii (np. do rozmiaru 100x100) - ale z zachowaniem sensownych proporcji (nie muszą być to same kwadraty). Ważne natomiast jest to, by na koniec każda fotografia miała ten sam rozmiar.
- Konwertujemy je do skali szarości.
- Zamieniamy obrazy (tablice dwuwymiarowe, *de facto* macierze) na wektory poprzez odczytanie kolejno wartości poszczególnych pikseli (wtedy np. fotografia 120x80 pikseli staje się wektorem 9600-wymiarowym). Każdy taki wektor będzie stanowił jedną obserwację.

No dobra, pora się wreszcie pobawić PCA

Przechodzimy teraz do części właściwej zadania. Na tak przygotowanym zbiorze danych wykonujemy transformację PCA, wcześniej go oczywiście centrując (gotowa implementacja np. tutaj <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>). Następnie staramy się odpowiedzieć na poniższe pytania (czasem może być potrzebne sporządzenie wykresu lub niewielkiej wizualizacji).

- Jak wyglądała dla tego zbioru macierz kowariancji przed transformacją PCA? Jak po jej wykonaniu?
- Jak rozkładały się wariancje poszczególnych cech przed transformacją PCA? A jak po jej użyciu?
- Jak wyglądało średnie zdjęcie (to które odjęliśmy od pozostałych, by wycentrować zbiór)?
 - Do realizacji tego i kilku kolejnych punktów przyda się funkcja, która wykona operację odwrotną do wektoryzacji - czyli zamieni wektor (w tym przypadku zawierający średnie wartości poszczególnych cech) na fotografię.
- Jak wyglądają znalezione nowe wektory bazowe (czyli główne składowe – *principal components*)? Zaprezentuj je posortowane według powiązanej wariancji.
 - Zauważmy, że wektory bazowe też są wektorami z oryginalnej przestrzeni. A że oryginalna przestrzeń zawierała fotografie, to znalezionej "lepszą" bazę możemy również zwizualizować w postaci obrazów, tak jak średnią fotografię z poprzedniego punktu.
- Zredukujemy wymiarowość naszych obserwacji do odpowiednio 3, 9 i 27 najważniejszych cech. Jak wyglądają tak "odchudzone" z wymiarów fotografie? Żeby odpowiedzieć na to pytanie wykonaj poniższe kroki.
 - Wyzerujemy wartości wszystkich cech poza tą wybraną garstką.
 - Przetransformujemy tak zmodyfikowane obserwacje ponownie do oryginalnej bazy (może być konieczne użycie odwrotności macierzy przejścia lub odpowiedniej metody z bibliotecznej implementacji).
 - Dodajmy do każdej z nich średni wektor (odwracając wycentrowanie).
 - Przekształćmy wektor ponownie do kształtu fotografii i wyświetlmy.
 - W praktyce wygląda to tak: robimy PCA (konwersja: fotografia → wektor cech w nowej bazie), usuwamy zbędne cechy (zerując pozostałe), robimy odwrotność PCA (konwersja: zmodyfikowany wektor cech w nowej bazie → fotografia).
- Na koniec użyjmy PCA do rzutowania naszego zbioru na płaszczyznę 2D. Jak wygląda taka wizualizacja? Tym razem trzeba będzie wykonać nieco inne kroki.
 - Zredukujemy wymiarowość do 2 najważniejszych aspektów danych. Nie zerujemy odrzucanych cech, a po prostu odpowiednio skróćmy wektory (obserwacje powinny stać się wektorami dwuwymiarowymi).
 - Użyjmy tych wektorów 2D jako współrzędnych na płaszczyźnie. Dla każdej z obserwacji namalujmy w tym miejscu kropkę. Uzależnijmy kolor danej kropki od typu obiektu, który reprezentowała obserwacja (np. spódnica → niebieska kropka, sukienka → czerwona kropka, etc.).
 - Alternatywnie (jeżeli umiemy sprawnie operować biblioteką do tworzenia wykresów) możemy umieścić na tych współrzędnych nie kropkę, a niewielką miniaturkę oryginalnej fotografii. ;]